ITS Technical Bulletin # 305

FAULT ANALYZER

Issued Date: October 1, 2002
Effective Date: October 1, 2002
Section/Group: Systems
Submitted By: Dennis Hansen
Approved by: Clair Christensen

In an effort to reduce software costs, and possibly increase the functionality of a software product, ITS is converting from ABEND Aid to Fault Analyzer. Both products are quite similar in functionality, but the cost savings will be considerable by converting to Fault Analyzer.

The Operating System running on the mainframes is developed by IBM, as is Fault Analyzer, therefore, maintenance may require fewer resources.

The purpose of Fault Analyzer is to determine why an application abends. After analyzing information about the application and its environment, Fault Analyzer generates an analysis report. The report describes the problem in terms of application code, which means that application developers and maintainers are not forced to interpret a low level system dump or system-level error message. As a result, the reason for the abend is made available sooner and with less effort.

When Fault Analyzer is attempting to analyze an abend, it looks for source line information to match with dump information. If Fault Analyzer cannot find the source, then it is not able to provide source line detail, though it can still provide an analysis of the abend.

To create a source file (or *side file*, as Fault Analyzer refers to such a listing file), a program named IDILANGX is used.

Fault Analyzer has been set up to look for side files named XX.DUMPSIDE, where XX is the assigned agency code which should be the first two characters of the jobname.

Please note: When creating side files using IDILANGX, you may NOT specify a member name on the DD statement that identifies the PDS(E) side file data set.

An example:

```
//CBLRUN   EXEC IGYWC,PARM.COBOL='LIST,MAP,SOURCE,XREF'
//COBOL.SYSIN DD *
.
```

. source code
.

```
//COBOL.SYSPRINT DD DSN=&&COBLIST(IDISCBL1),
//        DISP=(,PASS),SPACE=(TRK,(10,5,5),RLSE),
//        DCB=(RECFM=FBA,LRECL=133,BLKSIZE=0)
//*
//IDILANGX EXEC PGM=IDILANGX,REGION=4096K,
// PARM='IDISCBL1 (COBOL ERROR OFT IDILANGX FAULT'
//LISTING  DD  DISP=(OLD,PASS),DSN=&&COBLIST
//IDILANGX DD  DISP=SHR,DSN=XX.DUMPSIDE
//SYSUDUMP DD  SYSOUT=*
//*
```

If you already have listings, you can easily turn them into side files.

Here is an example:

```
//IDILANGX EXEC PGM=IDILANGX,REGION=4096K,
// PARM='COBTEST1  (COBOL ERROR OFT IDILANGX FAULT'
//LISTING  DD  DISP=SHR,DSN=XX.COBOL.LISTINGS
//IDILANGX DD  DISP=SHR,DSN=XX.DUMPSIDE                    (Side file)
//SYSUDUMP DD  SYSOUT=*
```

The Parm field is comprised as follows:

PARM=’mbr_name (language ERROR OFT langxdd FAULT’

where:

mbr_name — is the source listing member name in the input data set identified by the //LISTING DD name.

language — is the language of the compiler listing, as:

- COBOL
- PL1
- C
- ASM

langxdd — is the DD name used to identify the output side file data set.

The FAULT option may be abbreviated to FA.

Fault Analyzer has the ability to break abnormal termination dumps into data sets unique to the agency codes that have been assigned. The determination of which data set is to be used to store dumps is based upon the first two characters of the jobname, i.e., //SWDUMP JOB ... would be directed to the SW.DUMPHIST data set.

ITS will implement this feature to prevent unauthorized users from viewing sensitive data contained in dumps from other agencies.

Each agency is responsible for setting up their own dump data sets.

A XX.DUMPHIST data set may be allocated using the main ISPF 3.2 Option, where XX is the two character agency code assigned by accounting. Following is an example of a data set created with its parameters:

Data Set Name	SW.DUMPHIST	
Management Class	MGMTTSO	(Blank for default management class)
Storage Class	STANDARD	(Blank for default storage class)
Volume Serial		(Blank for system default volume) **
Device Type		(Generic unit or device address) **
Data Class		(Blank for default data class)
Space Units	CYLINDER	(BLKS, TRKS, CYLS, KB, MB, BYTES or RECORDS)
Average Record Unit		(M, K, or U)
Primary Quantity	10	(In above units)
Secondary Quantity	10	(In above units)
Directory Blocks	10	(Zero for sequential data set) *
Record Format	VB	
Record Length	10000	
Block Size		
Data Set Name Type	PDS	(LIBRARY, HFS, PDS, or blank) *
		(YY/MM/DD, YYYY/MM/DD

If a data set *has* been allocated when a dump is taken, the following statements will appear in the JES output for the job:

```
IEE252I MEMBER IDICNF00 FOUND IN SYS1.PARMLIB
+IDI0003I Fault ID F00001 assigned in history file SW.DUMPHIST
IEA995I SYMPTOM DUMP OUTPUT
```

```
SYSTEM COMPLETION CODE=0C7  REASON CODE=00000000
 TIME=14.51.35  SEQ=00919  CPU=0000  ASID=0052
 PSW AT TIME OF ERROR  078D0000   0000779A  ILC 6  INTC 07
   ACTIVE LOAD MODULE          ADDRESS=00007550  OFFSET=0000024A
   NAME=CWABTEST
```

The SW.DUMPHIST will of course be replaced with *your* allocated dump data set name.

If *your* data set *has not* been allocated, no dump will be recorded and you will receive messages in the JES output log to indicate that, as follows:

```
IEE252I MEMBER IDICNF00 FOUND IN SYS1.PARMLIB
+IDI0078E Open of fault history file 'SW.DUMPHIST' failed because: EDC5049I The
specified file name could not be located.
+IDI0053I Fault history file entry suppressed
+IDI0002I A Data Exception occurred in module CWABTEST CSECT CWABTEST
 at offset X'244' that was caused by a ADD DECIMAL machine instruction
IEA995I SYMPTOM DUMP OUTPUT
SYSTEM COMPLETION CODE=0C7  REASON CODE=00000000
 TIME=14.55.10  SEQ=00924  CPU=0000  ASID=0052
 PSW AT TIME OF ERROR  078D0000   0000779A  ILC 6  INTC 07
   ACTIVE LOAD MODULE          ADDRESS=00007550  OFFSET=0000024A
   NAME=CWABTEST
```

Each agency will be responsible for its own DUMPHIST data sets. This includes allocating the data set, allocating enough space to contain a certain number of dumps, and maintaining the data set by deleting unneeded dumps. If a dump data set fills up, a new bigger data set may be allocated and dumps from the older data set may be copied into the new one.

When a dump has been recorded into *your* XX.DUMPHIST data set you may view it by choosing option FA from the main ISPF panel on your mainframe terminal.

You will be asked to supply the name of *your* dump data set:

Fault History File or View : 'SW.DUMPHIST' <==== Change this name

Fault Analyzer will then display the available dumps for analysis.

Following is the two character agency code list followed by the dump data set name, where a dump will be directed, based on the jobname, within Fault Analyzer:

'AC'
  'AC.DUMPHIST'
'AD'
  'AD.DUMPHIST'
'AG'
  'AG.DUMPHIST'
'AR'
  'AR.DUMPHIST'
'AS'
  'AS.DUMPHIST'
'AT'
  'AT.DUMPHIST'
'BD'
  'BD.DUMPHIST'
'BE'
  'BE.DUMPHIST'
'AC'
  'AC.DUMPHIST'
'BR'
  'BR.DUMPHIST'
'BS'
  'BS.DUMPHIST'
'CA'
  'CA.DUMPHIST'
'CC'
  'CC.DUMPHIST'
'CD'
  'CD.DUMPHIST'
'CE'
  'CE.DUMPHIST'
'CICSAT'
  'CICS.DUMPHIST.CICSAT'
'CICSCJ'
  'CICS.DUMPHIST.CICSCJ'
'CICSDV'
  'CICS.DUMPHIST.CICSDV'
'CICSEP'
  'CICS.DUMPHIST.CICSEP'
'CICSGG'
  'CICS.DUMPHIST.CICSGG'
'CICSHS'

'CICS.DUMPHIST.CICSHS'
'CICSIT'
'CICS.DUMPHIST.CICSIT'
'CICSM'
'CICS.DUMPHIST.CICSM'
'CICSOV'
'CICS.DUMPHIST.CICSOV'
'CICSPA'
'CICS.DUMPHIST.CICSPA'
'CICSRA'
'CICS.DUMPHIST.CICSRA'
'CICSRD'
'CICS.DUMPHIST.CICSRD'
'CICSRG'
'CICS.DUMPHIST.CICSRG'
'CICSRM'
'CICS.DUMPHIST.CICSRM'
'CICSRS'
'CICS.DUMPHIST.CICSRS'
'CICSRT'
'CICS.DUMPHIST.CICSRT'
'CICSRX'
'CICS.DUMPHIST.CICSRX'
'CICSTR'
'CICS.DUMPHIST.CICSTR'
'CICST1'
'CICS.DUMPHIST.CICST1'
'CICST2'
'CICS.DUMPHIST.CICST2'
'CICST3'
'CICS.DUMPHIST.CICST3'

'CICST'
'CICS.DUMPHIST.CICST'
'CICSWS'
'CICS.DUMPHIST.CICSWS'
'CICSW1'
'CICS.DUMPHIST.CICSW1'
'CICSW2'
'CICS.DUMPHIST.CICSW2'
'CI'

'CI.DUMPHIST'
'CJ'
'CJ.DUMPHIST'
'CO'
'CO.DUMPHIST'
'CP'
'CP.DUMPHIST'
'CR'
'CR.DUMPHIST'
'CS'
'CS.DUMPHIST'
'DB'
'DB.DUMPHIST'
'DL'
'DL.DUMPHIST'
'DP'
'DP.DUMPHIST'
'EQ'
'EW.DUMPHIST'
'ES'
'ES.DUMPHIST'
'FC'
'FC.DUMPHIST'
'FD'
'FD.DUMPHIST'
'FI'
'FI.DUMPHIST'
'FM'
'FM.DUMPHIST'
'FO'
'FO.DUMPHIST'
'GP'
'GP.DUMPHIST'

'GR'
'GR.DUMPHIST'
'GV'
'GV.DUMPHIST'
'HC'
'HC.DUMPHIST'
'HE'

'HE.DUMPHIST'
'HL'
'HL.DUMPHIST'
'HP'
'HP.DUMPHIST'
'HS'
'HS.DUMPHIST'
'IC'
'IC.DUMPHIST'
'ID'
'ID.DUMPHIST'
'IF'
'IF.DUMPHIST'
'IR'
'IR.DUMPHIST'
'JI'
'JI.DUMPHIST'
'LA'
'LA.DUMPHIST'
'LE'
'LE.DUMPHIST'
'LG'
'LG.DUMPHIST'
'LH'
'LH.DUMPHIST'
'LI'
'LI.DUMPHIST'
'LQ'
'LQ.DUMPHIST'
'LR'
'LR.DUMPHIST'
'LS'
'LS.DUMPHIST'
'LV'
'LV.DUMPHIST'

'ME'
'ME.DUMPHIST'
'MI'
'MI.DUMPHIST'
'NA'

'NA.DUMPHIST'
'NG'
'NG.DUMPHIST'
'NR'
'NR.DUMPHIST'
'NS'
'NS.DUMPHIST'
'NT'
'NT.DUMPHIST'
'OE'
'OE.DUMPHIST'
'OR'
'OR.DUMPHIST'
'PA'
'PA.DUMPHIST'
'PE'
'PE.DUMPHIST'
'PR'
'PR.DUMPHIST'
'PS'
'PS.DUMPHIST'
'PU'
'PU.DUMPHIST'
'RB'
'RB.DUMPHIST'
'RE'
'RE.DUMPHIST'
'RG'
'RG.DUMPHIST'
'RM'
'RM.DUMPHIST'
'SA'
'SA.DUMPHIST'
'SD'
'SD.DUMPHIST'
'SE'
'SE.DUMPHIST'

'SP'
'SP.DUMPHIST'
'SR'

'SR.DUMPHIST'
'SS'
'SS.DUMPHIST'
'ST'
'ST.DUMPHIST'
'SW'
'SW.DUMPHIST'
'SY'
'SY.DUMPHIST'
'TA'
'TA.DUMPHIST'
'TC'
'TC.DUMPHIST'
'TL'
'TL.DUMPHIST'
'TV'
'TV.DUMPHIST'
'TX'
'TX.DUMPHIST'
'UC'
'UC.DUMPHIST'
'UI'
'UI.DUMPHIST'
'UL'
'UL.DUMPHIST'
'US'
'US.DUMPHIST'
'WE'
'WE.DUMPHIST'
'WF'
'WF.DUMPHIST'
'WP'
'WP.DUMPHIST'
'WS'
'WS.DUMPHIST'
'WU'
'WU.DUMPHIST'

All other dumps will sent to the ‘DP.DUMPHIST.DEFAULT’ default dump data set.